

⑥

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-033229

(43)Date of publication of application : 09.02.1999

(51)Int.Cl.

A63F 9/22

B06B 1/04

(21)Application number : 09-192166

(71)Applicant : NINTENDO CO LTD

(22)Date of filing : 17.07.1997

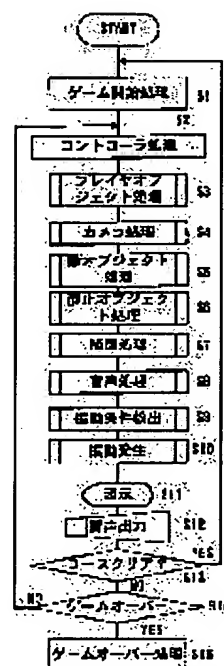
(72)Inventor : SAIKAI SATOSHI
KOUJIMA KAZUO
OTA KEIZO

(54) VIDEO GAME SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To play a video game with a more actual feeling by giving a player a vibration in accordance with a game picture.

SOLUTION: CPU detect a vibration generating condition such as the collision of contact of a player object against/with another object for example, by a step S9 based on the game picture (a player object picture, an enemy object picture or a still object picture) which is processed by a step S7. When the vibration generating condition is detected, a vibration source included in a vibration cartridge is driven so as to generate the vibration in a step S10.



LEGAL STATUS

[Date of request for examination]

23.05.2003

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

THIS PAGE BLANK (USPTO)

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平11-33229

(43)公開日 平成11年(1999) 2月9日

(51)Int.Cl.⁹

識別記号

F I

A 6 3 F 9/22

A 6 3 F 9/22

F

A

H

B 0 6 B 1/04

B 0 6 B 1/04

A

審査請求 未請求 請求項の数15 O L (全 20 頁)

(21)出願番号

特願平9-192166

(22)出願日

平成9年(1997) 7月17日

(71)出願人 000233778

任天堂株式会社

京都府京都市東山区福稲上高松町60番地

(72)発明者 西海 聡

京都府京都市東山区福稲上高松町60番地

任天堂株式会社内

(72)発明者 幸嶋 一雄

京都府京都市東山区福稲上高松町60番地

任天堂株式会社内

(72)発明者 太田 敬三

京都府京都市東山区福稲上高松町60番地

任天堂株式会社内

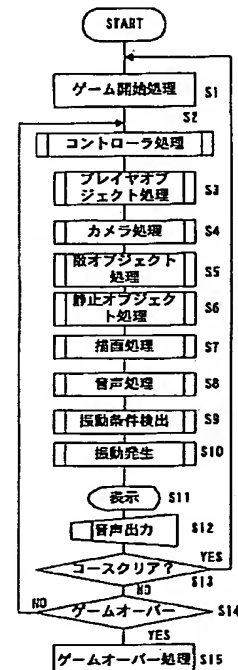
(74)代理人 弁理士 山田 義人

(54)【発明の名称】 ビデオゲームシステム

(57)【要約】

【構成】 CPUは、ステップS7によって処理されたゲーム画像（プレイヤオブジェクト画像、敵オブジェクト画像および静止オブジェクト画像）に基づいて、たとえばプレイヤオブジェクトが他のオブジェクトと衝突しない接触する等した、振動発生条件をステップS9で検出する。振動発生条件が検出されると、ステップS10において、振動カートリッジに含まれる振動源を駆動し、振動を発生させる。

【効果】 ゲーム画像に応じてプレイヤに振動を与えることができ、より一層実感を伴ってビデオゲームをプレイすることができる。



【特許請求の範囲】

【請求項 1】ビデオゲーム機と記憶媒体とコントローラとを備え、前記ビデオゲーム機は前記コントローラの操作状態と前記記憶媒体に記憶されたゲームプログラムとに基づいてディスプレイに表示されるべきゲーム画像を発生する処理手段を含む、ビデオゲームシステムであって、

前記コントローラは、プレイヤーの手によって把持された状態で使用され、さらに

プレイヤーオブジェクトの移動または動作を指示する複数の操作スイッチと、

機械的振動を発生する振動源と、

前記ビデオゲーム機からの指令信号にตอบสนองして前記振動源を駆動する駆動回路とを備え、

前記記憶媒体は、

前記操作スイッチの少なくとも 1 つの操作に応じて変化するプレイヤーオブジェクトの画像を発生するプレイヤーオブジェクト画像発生プログラムと、

前記プレイヤーオブジェクトの周囲に存在する少なくとも 1 つの他のオブジェクト画像を発生する他のオブジェクト画像発生プログラムと、

前記プレイヤーオブジェクト画像と前記他のオブジェクト画像の表示状態に基づいて振動発生条件を検出する振動発生条件検出プログラムと、

前記振動発生条件検出プログラムによって振動発生条件を検出したことにตอบสนองして前記振動源を制御するためのデータを発生する振動制御プログラムとを含み、

前記処理手段は、

前記プレイヤーオブジェクト画像発生プログラムと前記他のオブジェクト画像発生プログラムとに基づいて前記ゲーム画像を発生するとともに、

前記振動制御プログラムに基づいて前記ディスプレイのフレームに同期したタイミングで前記駆動回路によって前記振動源に機械的な振動を発生させる、ビデオゲームシステム。

【請求項 2】前記振動発生条件検出プログラムは複数種類の振動発生条件を検出するようにプログラムされていて、

前記振動制御プログラムは前記振動発生条件検出プログラムによって検出される異なる振動発生条件に応じて異なる制御データを発生する、請求項 1 記載のビデオゲームシステム。

【請求項 3】前記振動発生条件検出プログラムは、前記プレイヤーオブジェクト画像と前記他のオブジェクト画像との関係に従って弱い振動を発生すべき第 1 の振動発生条件と、前記関係に従って強い振動を発生すべき第 2 の振動発生条件とを検出するプログラムを含み、

前記処理手段は、前記振動発生条件検出プログラムによって前記第 1 の振動発生条件が検出されたことにตอบสนองして前記振動源が相対的に弱い振動を発生するように前記

駆動回路を制御し、第 2 の振動発生条件が検出されたことにตอบสนองして前記振動源が相対的に強い振動を発生するように前記駆動回路を制御する、請求項 1 または 2 に記載のビデオゲームシステム。

【請求項 4】前記振動制御プログラムは、前記振動源をオンまたはオフするフレーム数の違いに応じて前記振動源によって前記相対的に強い振動または前記相対的に弱い振動を発生させる間欠制御プログラムである、請求項 3 記載のビデオゲームシステム。

【請求項 5】前記振動制御プログラムは、前記プレイヤーオブジェクト画像および前記他のオブジェクト画像の表示状態に基づいて、前記振動源をオンまたはオフする前記フレーム数を算出する、請求項 4 記載のビデオゲームシステム。

【請求項 6】前記振動制御プログラムは、前記プレイヤー操作オブジェクトと前記他のオブジェクトの表示状態によって予め定められた複数の振動パターンデータを含む、請求項 2 記載のビデオゲームシステム。

【請求項 7】前記記憶媒体は、前記振動発生条件検出プログラムによって振動発生条件が検出されたとき、前記プレイヤーオブジェクトおよび前記他のオブジェクトの少なくとも一方の画像を視覚的に振動させる画像振動プログラムを含む、請求項 1 記載のビデオゲームシステム。

【請求項 8】前記画像振動プログラムは前記振動源による機械的振動の発生タイミングよりも遅れて画像を振動させる、請求項 6 記載のビデオゲームシステム。

【請求項 9】ビデオゲーム機とそれに操作信号を与えるコントローラとを備える、ビデオゲームシステムであって、

前記コントローラはプレイヤーの手によって把持された状態で使用され、さらにプレイヤーオブジェクトの移動または動作を指示する複数の操作スイッチと、機械的振動を発生する振動源と、前記ビデオゲーム機からの指令信号にตอบสนองして前記振動源を駆動する駆動回路とを備え、前記ビデオゲーム機は

前記操作スイッチの少なくとも 1 つの操作に応じて変化するプレイヤーオブジェクトの画像を発生するプレイヤーオブジェクト画像発生手段、

前記プレイヤーオブジェクトの周囲に存在する少なくとも 1 つの他のオブジェクト画像を発生する他のオブジェクト画像発生手段、

前記プレイヤーオブジェクト画像と前記他のオブジェクト画像の表示状態に基づいて振動発生条件を検出する振動発生条件検出手段、および前記振動発生条件検出手段によって前記振動発生条件が検出されたことにตอบสนองしてディスプレイのフレームに同期したタイミングで前記駆動回路によって前記振動源に機械的な振動を発生させる振動発生手段を含む、ビデオゲームシステム。

【請求項 10】前記振動発生条件検出手段は複数種類の振動発生条件を検出し、

3

前記振動発生手段は前記振動発生条件検出手段によって検出される異なる振動発生条件に応じて異なる態様で前記振動源を駆動する駆動手段を含む、請求項 9 記載のビデオゲームシステム。

【請求項 1 1】前記振動発生条件検出手段は、前記プレイヤーオブジェクト画像と前記他のオブジェクト画像との関係に従って弱い振動を発生すべき第 1 の振動発生条件を検出する第 1 検出手段と、前記関係に従って強い振動を発生すべき第 2 の振動発生条件を検出する第 2 検出手段を含む、

前記駆動手段は、前記第 1 検出手段によって前記第 1 の振動発生条件が検出されたことに応答して前記振動源が相対的に弱い振動を発生するように前記駆動回路を制御し、前記第 2 検出手段によって前記第 2 の振動発生条件が検出されたことに応答して前記振動源が相対的に強い振動を発生するように前記駆動回路を制御する、請求項 1 0 記載のビデオゲームシステム。

【請求項 1 2】前記駆動手段は前記駆動回路によって前記振動源を間欠的にオンまたはオフさせる間欠制御手段を含む、請求項 1 1 記載のビデオゲームシステム。

【請求項 1 3】前記間欠制御手段は前記振動源をオンまたはオフするフレーム数を算出する算出手段を含む、請求項 1 2 記載のビデオゲームシステム。

【請求項 1 4】ビデオゲーム機と記憶媒体とコントローラとを備え、前記ビデオゲーム機は前記コントローラの操作状態と前記記憶媒体に記憶されたゲームプログラムとに基づいてディスプレイに表示されるべきゲーム画像を発生する処理手段を含む、ビデオゲームシステムであって、

前記コントローラは、プレイヤーの手によって把持された状態で使用され、さらにプレイヤーオブジェクトの移動または動作を指示する複数の操作スイッチと、

機械的振動を発生する振動源と、

前記ビデオゲーム機からの指令信号に応じて前記振動源を駆動する駆動回路とを備え、

前記記憶媒体は、

前記操作スイッチの少なくとも 1 つの操作に応じて変化するプレイヤーオブジェクトの画像を発生するプレイヤーオブジェクト画像発生プログラムと、

前記プレイヤーオブジェクトの周囲に存在する少なくとも 1 つの他のオブジェクト画像を発生する他のオブジェクト画像発生プログラムと、

前記コントローラの前記操作スイッチの操作状態に基づいて振動発生条件を検出する振動発生条件検出プログラムと、

前記振動発生条件検出プログラムによって振動発生条件を検出したことに応答して前記振動源を制御するためのデータを発生する振動制御プログラムとを含み、

前記処理手段は、

前記プレイヤーオブジェクト画像発生プログラムと前記他

4

のオブジェクト画像発生プログラムとに基づいて前記ゲーム画像を発生するとともに、

前記振動制御プログラムに基づいて前記ディスプレイのフレームに同期したタイミングで前記駆動回路によって前記振動源に機械的な振動を発生させる、ビデオゲームシステム。

【請求項 1 5】ビデオゲーム機とそれに操作信号を与えるコントローラとを備える、ビデオゲームシステムであって、

10 前記コントローラはプレイヤーの手によって把持された状態で使用され、さらにプレイヤーオブジェクトの移動または動作を指示する複数の操作スイッチと、機械的振動を発生する振動源と、前記ビデオゲーム機からの指令信号に応じて前記振動源を駆動する駆動回路とを備え、

前記ビデオゲーム機は前記操作スイッチの少なくとも 1 つの操作に応じて変化するプレイヤーオブジェクトの画像を発生するプレイヤーオブジェクト画像発生手段、

前記プレイヤーオブジェクトに作用する前記操作スイッチの操作状態に基づいて振動発生条件を検出する振動発生条件検出手段、および前記振動発生条件検出手段によっ

20 て前記振動発生条件が検出されたことに応答してディスプレイのフレームに同期したタイミングで前記駆動回路によって前記振動源に機械的な振動を発生させる振動発生手段を含む、ビデオゲームシステム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明はビデオゲームシステムに関する。より特定的には、この発明は、ビデオゲーム機に操作信号を与えるコントローラに設けられた振動源によってゲーム画像に応じて振動を発生させる、新規なビデオゲームシステムに関する。

【0002】

【従来の技術】従来のいわゆる業務用ビデオゲーム機では、プレイヤーの座るシートに振動を発生させたり、あるいはそのシートを傾けたりして、ゲームの興趣を一層た高めることができる、いわゆる体感ゲーム機が既に実用化されている。また、たとえば昭和 6 2 年（1987）12 月 10 日付で出願公開された実開昭 6 2 - 194389 号には、ジョイスティックレバー内に電磁石を内蔵し、その電磁石によってジョイスティックレバーに振動を発生させるコントローラが開示されている。このような振動発生可能なコントローラを家庭用ビデオゲーム機に接続することによって、プレイヤーの手に振動が伝達されるゲームを構成することが考えられる。

【0003】

【発明が解決しようとする課題】しかしながら、上述の従来技術は、ディスプレイに表示されるゲーム画像と振動との関連が全く開示されておらず、したがって、この従来技術を家庭用ビデオゲームシステムに利用することはできない。また、上述の体感ゲーム機は、シートに振

動を発生させるものであるため、そのようなシートを持たない家庭用ビデオゲームシステムでは利用できない。

【0004】それゆえに、この発明の主たる目的は、ゲーム画像と関連してプレイヤに振動を伝達することができる、ビデオゲームシステムを提供することである。

【0005】

【課題を解決するための手段】この発明は、ビデオゲーム機と記憶媒体とコントローラとを備え、ビデオゲーム機はコントローラの操作状態と記憶媒体に記憶されたゲームプログラムとに基づいてディスプレイに表示されるべきゲーム画像を発生する処理手段を含む、ビデオゲームシステムであって、コントローラは、プレイヤの手によって把持された状態で使用され、さらにプレイヤオブジェクトの移動または動作を指示する複数の操作スイッチと、機械的振動を発生する振動源と、ビデオゲーム機からの指令信号にตอบสนองして振動源を駆動する駆動回路とを備え、記憶媒体は、操作スイッチの少なくとも1つの操作に応じて変化するプレイヤオブジェクトの画像を発生するプレイヤオブジェクト画像発生プログラムと、プレイヤオブジェクトの周囲に存在する少なくとも1つの他のオブジェクト画像を発生する他のオブジェクト画像発生プログラムと、プレイヤオブジェクト画像と他のオブジェクト画像の表示状態に基づいて振動発生条件を検出する振動発生条件検出プログラムと、振動発生条件検出プログラムによって振動発生条件を検出したことにตอบสนองして振動源を制御するためのデータを発生する振動制御プログラムとを含み、処理手段は、プレイヤオブジェクト画像発生プログラムと他のオブジェクト画像発生プログラムとに基づいてゲーム画像を発生するとともに、振動制御プログラムに基づいてディスプレイのフレームに同期したタイミングで駆動回路によって振動源に機械的な振動を発生させる、ビデオゲームシステムである。

【0006】この発明は、ビデオゲーム機とそれに操作信号を与えるコントローラとを備える、ビデオゲームシステムであって、コントローラはプレイヤの手によって把持された状態で使用され、さらにプレイヤオブジェクトの移動または動作を指示する複数の操作スイッチと、機械的振動を発生する振動源と、ビデオゲーム機からの指令信号にตอบสนองして振動源を駆動する駆動回路とを備え、ビデオゲーム機は操作スイッチの少なくとも1つの操作に応じて変化するプレイヤオブジェクトの画像を発生するプレイヤオブジェクト画像発生手段、プレイヤオブジェクトの周囲に存在する少なくとも1つの他のオブジェクト画像を発生する他のオブジェクト画像発生手段、プレイヤオブジェクト画像と他のオブジェクト画像の表示状態に基づいて振動発生条件を検出する振動発生条件検出手段、および振動発生条件検出手段によって振動発生条件が検出されたことにตอบสนองしてディスプレイのフレームに同期したタイミングで駆動回路によって振動源に機械的な振動を発生させる振動発生手段を含む、ビ

デオゲームシステムである。

【0007】

【作用】プレイヤオブジェクト画像および他のオブジェクト画像の状態（たとえば、両者が当たっているかなど）に応じて、振動発生条件が検出される。この条件が検出されると、コントローラに含まれる振動源を駆動回路によって駆動する。また、コントローラの操作スイッチの操作状態によって、プレイヤオブジェクト画像と他のオブジェクト画像との関係によらず、振動を発生させることもできる。

【0008】

【発明の効果】この発明によれば、ゲーム画像の状態に応じてコントローラの振動源によって振動を発生させることができるので、より一層実感を伴ってゲームをプレイすることができる。この発明の上述の目的、その他の目的、特徴および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなろう。

【0009】

【実施例】図1に示す実施例のビデオゲームシステムは、ビデオゲーム機10と、記憶媒体の一例のROMカートリッジ20と、ビデオゲーム機10に接続されるディスプレイ30と、コントローラ40とを含んで構成される。コントローラ40には、振動カートリッジ50が着脱自在に装着される。

【0010】コントローラ40は、両手または片手で把持可能な形状のハウジング41に、複数のスイッチないしボタンを設けて構成される。具体的には、コントローラ40は、ハウジング41の左右端部および中央部に、それぞれ下方に延びて形成されるハンドル41L、41Cおよび41Rを含み、ハウジング41の上面が操作領域である。操作領域には、中央下部にアナログ入力可能なジョイスティック（以下、「アナログジョイスティック」という。）45が設けられ、左側に十字形のディジタル方向スイッチ（以下、「十字スイッチ」という。）46が設けられ、右側に複数のボタンスイッチ47A、47B、47C、47D、47Eおよび47Fが設けられる。

【0011】アナログジョイスティック45は、スティックの傾き量と方向とによって、プレイヤオブジェクト（プレイヤがコントローラ40によって操作可能なオブジェクト）の移動方向および/または移動速度ないし移動量を入力するために用いられる。十字スイッチ46は、アナログジョイスティック45に代えてプレイヤオブジェクトの移動方向を指示するために用いられる。ボタンスイッチ47Aおよび47Bは、プレイヤオブジェクトの動作を指示するために利用され、ボタンスイッチ47C-47Dは、三次元画像のカメラの視点を切り換えたり、プレイヤオブジェクトのスピード調節等に用いられる。

【0012】操作領域のほぼ中央部にはスタートスイッ

チ47Sが設けられ、このスタートスイッチ47Sは、ゲームを開始させるときに操作される。中央部のハンドル41Cの裏側にスイッチ47Zが設けられ、このスイッチ47Zは、たとえばシューティングゲームにおいてトリガスイッチとして利用される。ハウジング41の左右上部側面にはスイッチ47Lおよび47Rが設けられる。

【0013】なお、上述のボタンスイッチ47C-47Fは、カメラの視点切換え以外の用途として、シューティングまたはアクションゲームにおいてプレイャオブジェクトの移動速度を制御（たとえば、加速または減速）するためにも使用できる。しかしながら、これらのスイッチ47A-47F、47S、47Z、47Lおよび47Rの機能は、ゲームプログラムによって任意に定義することができる。

【0014】図2は図1実施例のビデオゲームシステムのブロック図である。ビデオゲーム機10には、中央処理ユニット（以下、「CPU」という。）11およびコプロセッサ（リアリティ・コプロセッサ；以下、「RCP」という。）12が内蔵される。RCP12には、バスの制御を行うためのバス制御回路121と、ポリゴンの座標変換や陰影処理等を行うための信号プロセッサ（リアリティ・シグナル・プロセッサ；以下、「RSP」という。）122と、ポリゴンデータを表示すべき画像にラスターライズしかつフレームメモリに記憶可能なデータ形式（ドットデータ）に変換するための描画プロセッサ（リアリティ・ディスプレイ・プロセッサ；以下、「RDP」という。）46とが含まれる。

【0015】RCP12には、外部ROM21を内蔵するROMカートリッジ20を着脱自在に装着するためのカートリッジ用コネクタ13と、ディスクドライブ29を着脱自在に装着するためのディスクドライブ用コネクタ197と、RAM14とが接続される。また、RCP12には、CPU11によって処理された音声信号および映像信号をそれぞれ出力するためのDAC（デジタル/アナログ変換器）15および16が接続される。さらに、RCP12には、1つまたは複数のコントローラ40の操作データおよび/または振動カートリッジ50のデータをシリアル転送するためのコントローラ制御回路17が接続される。

【0016】RCP12に含まれるバス制御回路121は、CPU11からバスを介してパラレル信号で与えられたコマンドをパラレル/シリアル変換して、シリアル信号としてコントローラ制御回路17に供給する。また、バス制御回路121は、コントローラ制御回路17から入力されたシリアル信号をパラレル信号に変換し、バスを介してCPU11へ出力する。コントローラ40から読み込まれた操作状態を示すデータ（操作信号ないし操作データ）は、CPU11によって処理されたり、RAM14に一時記憶される等の処理が行われる。換言

すれば、RAM14は、CPU11によって処理されるデータを一時記憶する記憶領域を含み、バス制御回路121を介してデータの読出または書込を円滑に行うことに利用される。

【0017】音声用DAC15には、ビデオゲーム機10の後面に設けられるコネクタ195が接続される。画像用DAC16には、ビデオゲーム機10の後面に設けられるコネクタ196が接続される。コネクタ195には、ディスプレイ30のスピーカ31が接続される。コネクタ196には、テレビジョン受像機またはCRT等のディスプレイ30が接続される。

【0018】コントローラ制御回路17には、ビデオゲーム機10の前面に設けられるコントローラ用コネクタ18が接続される。コネクタ18には、接続用ジャックを介してコントローラ40が着脱自在に接続される。このように、コネクタ18にコントローラ40を接続することにより、コントローラ40がビデオゲーム機10と電気的に接続され、相互間のデータの送受信または転送が可能とされる。

【0019】コントローラ制御回路17は、RCP12とコントローラ用コネクタ18との間でデータをシリアルで送受信するために用いられ、図3に示すように、データ転送制御回路171、送信回路172、受信回路173および送受信データを一時記憶するためのRAM174を含む。データ転送制御回路171は、データ転送時にデータフォーマットを変換するためにパラレル/シリアル変換回路とシリアル/パラレル変換回路を含み、さらにRAM174の書込/読出制御を行う。シリアル/パラレル変換回路は、RCP12から供給されるシリアルデータをパラレルデータに変換してRAM174または送信回路172に与える。パラレル/シリアル変換回路は、RAM174または受信回路173から供給されるパラレルデータをシリアルデータに変換して、RCP12に与える。送信回路172は、データ転送制御回路171から供給されるコントローラ40の信号読込のためのコマンドおよび振動カートリッジ50への書込データ（パラレルデータ）をシリアルデータに変換して、各コントローラ40のそれぞれに対応するチャンネルCH1~CH4へ送出する。受信回路173は、各コントローラ40に対応するチャンネルCH1~CH4から入力される各コントローラの操作データおよび振動カートリッジ50からの読出データをシリアルデータで受信し、パラレルデータに変換してデータ転送制御回路171に与える。データ転送制御回路171は、RCP12から転送されたデータまたは受信回路173で受信されたコントローラデータや振動カートリッジ50の読出データをRAM174に書込んだり、RCP12からの命令に基づいてRAM174のデータを読み出してRCP12へ転送する。

【0020】なお、RAM174は、図示を省略してい

るが、各チャンネルCH1～CH4毎の記憶場所を有し、各記憶場所に当該チャンネルのコマンド、送信データおよび/または受信データがそれぞれ記憶される。図4はコントローラ40および振動カートリッジ50の詳細な回路図である。コントローラ40のハウジングには、ジョイスティック45、各スイッチ46、47等の操作状態を検出しかつその検出データをコントローラ制御回路17へ転送するために、操作信号処理回路44等が内蔵される。操作信号処理回路44は、受信回路441、制御回路442、スイッチ信号検出回路443、カウンタ回路444、ジョイポート制御回路446、リセット回路447およびNORゲート448を含む。受信回路441は、コントローラ制御回路17から送信される制御信号や振動カートリッジ50への書込データ等のシリアル信号をパラレル信号に変換して制御回路442に与える。制御回路442は、コントローラ制御回路17から送信される制御信号がジョイスティック45のX、Y座標のリセット信号であるとき、リセット信号を発生してNORゲート448を介してカウンタ444内のX軸用カウンタ444XとY軸用カウンタ444Yの計数値を

リセット(0)させる。
 【0021】ジョイスティック45は、レバーの傾き方向のX軸方向とY軸方向に分解して傾き量に比例したパルス数を発生するように、X軸用とY軸用のフォトインタラプトを含み、それぞれのパルス信号をカウンタ444Xおよびカウンタ444Yに与える。カウンタ444Xは、ジョイスティック45がX軸方向に傾けられたとき、その傾き量に応じて発生されるパルス数を計数する。カウンタ444Yは、ジョイスティック45がY軸方向に傾けられたとき、その傾き量に応じて発生されるパルス数を計数する。したがって、カウンタ444Xとカウンタ444Yとの計数値によって決まるX軸とY軸の合成ベクトルによって、プレイヤオブジェクトまたは主人公キャラクタもしくはカーソルの移動方向と座標位置とが決定される。なお、カウンタ444Xおよびカウンタ444Yは、電源投入時にリセット信号発生回路447から与えられるリセット信号、またはプレイヤが所定の2つのスイッチを同時に押圧したときにスイッチ信号検出回路443から与えられるリセット信号によって、リセットされる。

【0022】スイッチ信号検出回路443は、制御回路442から一定周期(たとえばテレビジョンのフレーム周期である1/30秒間隔)で与えられるスイッチ状態を出力するためのコマンドに回答して、十字スイッチ46およびスイッチ47A～47Zの押圧状態によって変化する信号を読み込み、それを制御回路442へ与える。制御回路442は、コントローラ制御回路17からの操作状態データの読出指令信号に回答して、各スイッチ47A～47Zの操作状態データおよびカウンタ444Xおよび444Yの計数値を所定のデータフォーマットで

送信回路445に与える。送信回路445は、制御回路442から出力されたパラレル信号をシリアル信号に変換して、変換回路43および信号線42を介してコントローラ制御回路17へ転送する。制御回路442には、アドレスバスおよびデータバスならびにポートコネクタ46を介してジョイポート制御回路446が接続される。ジョイポート制御回路446は、振動カートリッジ50がポートコネクタ46に接続されているとき、CPU11の命令に従ってデータの入出力(または送受信)制御を行う。

【0023】振動カートリッジ50は、アドレスバスおよびデータバスにRAM51を接続し、RAM51に電池52を接続して構成される。RAM51は、たとえばアドレスバスを用いてアクセス可能な最大容量の半分未満の容量(たとえば256kビット)のRAMである。RAM51は、ゲームに関連するバックアップデータを記憶するものであり、振動カートリッジ50がポートコネクタ46から抜き取られても電池52からの電力供給を受けてバックアップデータを保持する。この振動カートリッジ50は、振動発生回路53を内蔵する。

【0024】次に、図5を参照して、振動カートリッジ50を詳細に説明する。振動カートリッジ50は、ケース501とそのケース501に取り付けられる裏蓋502とを含む。このケース501および裏蓋502で形成された振動カートリッジ50が、図1に示すコントローラ40の開口部に、着脱自在に装着される。ケース501内には、基板503が収納される。基板503上には、前述のRAM51およびバックアップ電池52の他に、図4の振動発生回路53を構成する、電池504および駆動回路505が実装される。なお、基板503の手前側エッジには、上述のコントローラ40の開口部に形成されたコネクタ(図示せず)に接続される複数の端子506が形成される。この端子506がビデオゲーム機10のCPU11(図2)すなわちコントローラ制御回路17からのデータおよびアドレスを受ける。

【0025】裏蓋502には、振動発生回路53を構成する振動源507が固着される。実施例では、振動源507として、振動発生用モータを用いた。しかしながら、モータの他に、電源の供給を受けて振動を発生するソレノイドや他の素子が利用可能であることは勿論である。なお、振動発生用モータの一例として、東京パーツ工業株式会社製の「FM16」、「FM23」、「FM25」あるいは「FM29」や「CM-5」等が利用可能である。「FM」モータでは、円筒型ケースに内蔵された回転軸に偏心部材を取り付け、回転軸の回転に応じて偏心部材が回転することによって、ケースに振動が生じる。「CM」モータでは、電機子コイル自体を偏心させて装着し、その電機子が回転することによって、振動を発生する。なお、ソレノイドを用いる場合には、ソレノイド内の磁心が往復運動することによって振動が発生

する。

【0026】いずれの場合も、このような振動源507は、電池504からの電源を受けて駆動回路505によって駆動され、振動を発生する。振動源507の消費電力は比較的大きく、したがって、この実施例では、RAM51のバックアップ電池52(図4)とは別に、電池504を設けた。したがって、電池504が消耗されたときには、裏蓋502に取り外し可能に設けられた電池蓋508を開いて、電池504を新しい電池に交換することができる。ただし、2つの電池52および504として、同じ電池を共通に用いることも、可能である。

【0027】また、コントローラケーブル(図示せず)に電源線を含ませ、その電源線によって、画像処理装置本体すなわちビデオゲーム機10から端子506を通して振動源507に電源を供給するようにしてもよい。その場合には、電源線の容量は、振動源507の必要電力を考慮して適宜選定されることはいうまでもない。さらに、この実施例では、振動源507で発生した振動が減衰することなくコントローラ40からプレイヤーの手に伝達され易いように、振動源507を裏蓋502に取り付けた。つまり、振動源507で発生した振動は、裏蓋502から、その裏蓋502に接触しているコントローラ40の開口部に伝達され、したがって、コントローラ40自体が振動する。そのため、コントローラ40を把持しているプレイヤーの手に、振動源507が発生した振動が伝達されることになる。したがって、振動源507の振動がコントローラ40を通してプレイヤーの手に伝達され得る限り、振動源507をケース501内の任意の位置に取り付けることができる。

【0028】なお、この実施例では、コントローラ40に振動カートリッジ50を装着することによってコントローラ40に振動源507を持たせた。しかしながら、振動カートリッジ50を用いることはなく、コントローラ40のハウジング内に、振動発生回路53(図4)すなわち振動源507、駆動回路505および電池504等を内蔵するようにしてもよい。

【0029】次に、図6を参照して、振動発生回路53を構成する駆動回路505を詳細に説明する。駆動回路505は、NANDゲート510から成るデコーダを含み、このNANDゲート510は、ビデオゲーム機10のCPU11(図2)からのアドレスデータA2~A14を、アドレスバスすなわち端子506(図5)を通して受ける。実施例のゲーム機システムにおいては、そのアドレスA0~A15の全てが「1」の場合、すなわち、アドレスFFFFFの範囲をCPU11が指定したとき、振動モードが設定され、振動源507を駆動するためのデータをCPU11から出力するようにしている。つまり、CPUアドレスFFFFFが指定されると、デコーダすなわちNANDゲート510の出力が「0」となる。このNANDゲート510の出力がNANDゲート

511に与えられる。NANDゲート511にはさらに、CPU11からの書込信号WEおよびチップイネーブル信号CEが与えられているので、NANDゲート511はNANDゲート510の出力および信号WEおよびCEに応答して、ラッチ512にラッチ信号を与える。したがって、ラッチ512は、CPU11がFFFFFアドレスを指定したとき、すなわち、振動モードにおいて、CPUデータD0を、データバスすなわち端子506を通して、ラッチする。このCPUデータD0は、振動源507で振動を発生すべきとき「1」として、振動を発生すべきでないとき「0」として出力される。ラッチ512の出力は抵抗513を介して、駆動トランジスタ514のベースに接続される。ラッチ512の出力が「1」のとき、トランジスタ514がオンし、「0」のときオフする。トランジスタ514がオンすると、電池504から振動源507(振動モータ)に駆動電流が流れ、駆動源507から振動が発生される。

【0030】図7はROMカートリッジ20(図1)に内蔵される外部ROM21のメモリ空間を示すメモリマップである。外部ROM21は、複数の記憶領域(以下、単に「領域」と呼ぶこともある)、たとえば図5に示すように、プログラム領域22、文字コード領域23、画像データ領域24およびサウンドメモリ領域25を含み、各種のプログラムを予め固定的に記憶している。

【0031】プログラム領域22は、ゲーム画像を処理するために必要なプログラムや、ゲーム内容に応じたゲームデータ等を記憶している。具体的には、プログラム領域22は、CPU11の動作プログラムを予め固定的に記憶するための記憶領域22a~22pを含む。メインプログラム領域22aには、後述の図9に示すゲーム等のメインルーチンの処理プログラムが記憶される。コントロールパッドデータ判断プログラム領域22bには、コントローラ40の操作データを処理するためのプログラムが記憶される。書込プログラム領域22cには、CPU11がRCP12に書込処理させるべきフレームメモリおよびZバッファへの書込プログラムが記憶される。たとえば、書込プログラム領域22cには、1つの背景画面で表示すべき複数の移動オブジェクトまたは背景オブジェクトのテクスチャデータに基づく画像データとして、色データをRAM14の画像データ領域201(図8)に書き込むプログラムが記憶される。移動プログラム領域22dには、CPU11がRCP12に作用して三次元空間中の移動物体の位置を変化させるための制御プログラムが記憶される。カメラ制御プログラム領域22eには、プレイヤーオブジェクトを含む移動オブジェクトや背景オブジェクトを三次元空間中のどの方向および/または位置で撮影させるかを制御するためのカメラ制御プログラムが記憶される。プレイヤーオブジェクトプログラム領域22fには、プレイヤーによって操作

されるオブジェクトの表示制御のためのプログラムが記憶される。敵オブジェクトプログラム領域22gには、プレイヤーオブジェクトに対して攻撃を加える敵オブジェクトの表示制御のためのプログラムが記憶される。背景プログラム領域22hには、CPU11がRCP12に作用して、三次元の背景面を作成させるための背景作成プログラムが記憶される。

【0032】文字コード領域23は、複数種類の文字コードを記憶する領域であって、たとえばコードに対応した複数種類の文字のドットデータを記憶している。文字コード領域23に記憶されている文字コードデータは、ゲームの進行においてプレイヤーに説明文を表示するために利用される。画像データ領域24は、背景オブジェクトおよび/または移動オブジェクトの各オブジェクト毎に複数のポリゴンの座標データおよびテクスチャデータ等の画像データをそれぞれ記憶するとともに、これらのオブジェクトを所定の位置に固定的に表示または移動表示させるための表示制御プログラムを記憶している。

【0033】サウンドメモリ領域25には、場面毎に対応して、その場面に適した上記メッセージを音声で出力するためのセリフや効果音やゲーム音楽等のサウンドデータが記憶される。なお、記憶媒体ないし外部記憶装置としては、ROMカートリッジ20に代えてまたはROMカートリッジ20に加えて、CD-ROMや磁気ディスク等の各種記憶媒体を用いてもよい。その場合、CD-ROMや磁気ディスク等の光学式または磁気式等のディスク状記憶媒体からゲームのための各種データ（プログラムデータおよび画像表示のためのデータを含む）を読み出しまたは必要に応じて書き込むために、ディスクドライブ29（図2）が設けられる。ディスクドライブ29は、外部ROM21と同様のプログラムデータが磁氣的または光学的に記憶された磁気ディスクまたは光ディスクに記憶されたデータを読み出し、そのデータをRAM14に転送する。

【0034】図8はRAM14のメモリ空間を示すメモリマップである。RAM15は、画像データ領域201およびプログラム領域202を含む。画像データ領域201は、図示しないが1フレーム分の画像データを一時記憶するフレームメモリ領域と、フレームメモリ領域のドット毎の奥行きデータを記憶するZバッファ領域とを含む。プログラムデータ領域202は、プログラムを一時的に記憶するための領域である。上述のROM21の各領域（図7）に設定されたプログラムデータが必要に応じてプログラムデータ領域202に一時的に記憶され、CPU11およびRCP12（図2）は、RAM14のプログラム領域をアクセスすることによって、ゲームを進行させる。同じように、画像データ領域201も、ROM21に記憶されている画像データを必要に応じて一時的に記憶しておくための領域であり、CPU11また

はRCP12によって直接アクセスされ得る。つまり、画像データ領域201は、外部ROM21に記憶されているゲーム画像表示のための静止オブジェクトおよび/または移動オブジェクトを構成する複数のポリゴンの座標データおよびテクスチャデータを記憶するものであって、画像処理動作に先立ってたとえば1コースまたはステージ分のデータが外部ROM21から転送される。

【0035】コントローラデータ記憶領域141は、コントローラ40から読み込まれた操作状態を示す操作状態データを一時記憶する。また、フラグ・レジスタ領域142は、CPU11がプログラムを実行中に、必要に応じて、フラグを設定し、あるいは変数または定数を記憶する。このフラグ・レジスタ領域143に設定されるフラグとしては、振動ゲームフラグF1、当たりフラグF2、前フレームフラグF3および振動フラグF4などがある。

【0036】振動ゲームフラグF1は、現在プレイしているゲームが振動カートリッジ50の振動源507によって振動を発生させる場面を含むかどうかを示すフラグであり、そのような場面があるとき「1」として、それ以外るとき「0」として設定される。当たりフラグF2には、後述する当たり判定ルーチン（図11）によって2つのオブジェクトが衝突ないし接触したとき「1」が設定され、それ以外るとき「0」が設定される。前フレームF3は、ディスプレイ30の1フレーム前のフレームにおいて上述の当たりフラグF2が「1」にセットされたかどうか、つまり直前のフレームにおいても2つのオブジェクトは衝突ないし接触したかどうかを設定するためのフラグである。前フレームにおいて当たり検出がなされたとき「1」が、それ以外るとき「0」がそれぞれ設定される。振動フラグF4は、振動源507によって振動を発生させるべきとき「1」として、それ以外るとき「0」として設定される。

【0037】変数n_addは、後述の振動発生条件検出ルーチン（図10）において、1フレーム毎にカウンタCTのカウンタ値nを増加させるための増分値であり、変数n_add_mは増分値n_addの変化値である。カウンタCTは、データサイズが32ビットであり、256進カウンタ（8ビットカウンタ）のように、そのカウンタ値nが「255」を超えたとき、振動源507（図5）を駆動して振動を発生させる。変数ax、ayおよびazは、プレイヤーオブジェクトのX軸、Y軸およびZ軸のそれぞれの加速度である。変数fx、fyおよびfzは、加速度ax、ayおよびazに定数accx、accyおよびacczをそれぞれ乗算した値である。

【0038】なお、変数n_addおよびn_add_mは、定数であってもよい。実施例では、変数n_addは、「255」または「150」に設定される。また、変数n_add_mは、たとえば、「20」または

15

「10」に設定される。図9はこの実施例のビデオゲームシステムのメインフロー図であり、電源が投入されると、最初のステップS1において、CPU11はスタートに際してビデオゲーム機10を所定の初期状態に設定する。たとえば、CPU11は、外部ROM21のプログラム領域22に記憶されているゲームプログラムのうちの立ち上げプログラムをRAM14のプログラム領域202に転送し、各パラメータを初期値に設定した後、図9の各ステップを順次実行する。

【0039】図9のメインフロー図の動作は、たとえば1フレーム（1/60秒）毎または2ないし3フレーム毎に行われるものであり、コースをクリアするまではステップS2～S14が繰り返し実行される。コースクリアに成功することなくゲームオーバーになると、ステップS14に続いて、ステップS15においてゲームオーバー処理が行われる。コースクリアに成功するとステップS13からステップS1へ戻る。

【0040】すなわち、ステップS1において、ゲームのコース画面および/またはコース選択画面の表示が行われるが、電源投入後にゲームを開始する場合は、最初のコース画面の表示が行われる。最初のコースをクリアすると、次のコースが設定される。ステップS1に続いて、ステップS2において、コントローラ処理が行われる。この処理は、コントローラ40のジョイスティック45、十字スイッチ46、およびスイッチ47A～47Zの何れが操作されたかを検出し、その操作状態の検出データ（コントローラデータ）を読み込み、読み込んだコントローラデータをRAM14のコントローラデータ領域141に書き込む。

【0041】ステップS3において、プレイヤーオブジェクトの表示のための処理が行われる。この処理は、基本的には、プレイヤーの操作するジョイスティック45の操作状態と敵からの攻撃の有無に基づいてその姿勢、方向、形状および位置を変化させる処理である。たとえば、外部ROM21の記憶領域22f（図7）から転送されたプログラムと記憶領域24から転送されたプレイヤーオブジェクトのポリゴンデータとコントローラデータすなわちジョイスティック45の操作状態とに基づいて、変化後のポリゴンデータを演算によって求める。その結果得られた複数のポリゴンにテクスチャデータによって色を付与する。

【0042】ステップS4において、カメラ処理が行われる。たとえば、カメラのファインダを通して見たときの視線または視界がプレイヤーがジョイスティック45によって指定したアングルとなるように、各オブジェクトに対する視点の座標を演算する。ステップS5において、敵オブジェクトの処理が行われる。この処理は、記憶領域22gおよび一部転送されたプログラムおよび記憶領域24（図2）から転送された敵オブジェクトのポリゴンデータに基づいて実行される。たとえば、プレイ

16

ャオブジェクトの動きを判断しながらプレイヤーオブジェクトに攻撃を加えたり進行を妨げる動きとなるように、敵オブジェクトの表示位置および/またはその形状をポリゴンデータの演算によって求めて、変化した敵オブジェクトの画像が表示される。これによって、敵オブジェクトは、プレイヤーオブジェクトに対して何らかの影響を与えるように働く。

【0043】ステップS6において、背景（または静止）オブジェクトの処理が行われる。この処理は、記憶領域22hから一部転送されたプログラムと記憶領域24（図2）から転送された静止オブジェクトのポリゴンデータとに基づいて、静止オブジェクトの表示位置およびその形状を演算する。ステップS7において、RSP122が描画処理を行う。すなわち、RCP12は、CPU11の制御の下に、RAM14の画像データ領域201に記憶されている敵オブジェクト、プレイヤーオブジェクト等の移動オブジェクトや背景等の静止オブジェクトのそれぞれのテクスチャデータに基づいて、移動オブジェクトおよび静止オブジェクトの表示のための画像データの変換処理（座標変換処理およびフレームメモリ描画処理）を行う。具体的には、複数の移動オブジェクトや静止オブジェクト毎の複数のポリゴンに色を付与する。

【0044】ステップS8において、CPU11がメッセージや音楽や効果音等の音声データに基づいて、音声処理を行なう。次のステップS9において、CPU11は、振動条件検出処理を行う。すなわち、このステップS9において、RCP12は、プレイヤーオブジェクトおよび敵オブジェクトもしくは静止オブジェクトの画像データに基づいて、もしくはコントローラ40からの操作データに基づいて、図10のサブルーチンを実行することによって、振動源507によって振動を発生させるための条件が成立したかどうかを検出する。ただし、振動条件検出サブルーチンについては、図10を参照して後に詳細に説明する。

【0045】次のステップS10において、CPU11は、ステップS9によって振動発生条件を検出したことに応答して、図12に示すサブルーチンを実行することによって、振動源507によって振動を発生させる。ステップS10もまた、図12を参照して後に詳細に説明する。ステップS11において、CPU11が、ステップS7において描画処理された結果により、RAM14のフレームメモリ領域に記憶されている画像データを読み出す。したがって、プレイヤーオブジェクト、移動オブジェクト、静止オブジェクトおよび敵オブジェクト等がディスプレイ30（図1、図2）の表示画面上に表示される。

【0046】ステップS12において、RCP12がステップS18において音声処理した結果得られる音声データを読み出すことにより、音楽および効果音または会

話等の音声出力させる。ステップS13において、コースをクリアしたか否かが判断（コースクリア検出）され、コースをクリアしていなければステップS14においてゲームオーバーになったか否かが判断され、ゲームオーバーでなければステップS2へ戻り、ゲームオーバーの条件が検出されるまでステップS2～S14が繰り返される。そして、プレイヤーに許容されているミス回数が所定の回数になるか、プレイヤーオブジェクトのライフを所定数量使い切る等のゲームオーバー条件になったことが検出されると、続くステップS15においてゲームの継続またはバックアップデータの記憶の選択等のゲームオーバー処理が行われる。

【0047】なお、ステップS13において、コースをクリアした条件（たとえば、ボスを倒す等）が検出されると、コースクリアの処理をした後、ステップS1へ戻る。図10を参照して、振動発生条件検出サブルーチンの最初のステップS101では、CPU11は、プレイヤーオブジェクトとそれに対する影響物体ないし障害物（他の移動オブジェクト、地面、海面、壁等の静止オブジェクト、敵オブジェクト、攻撃オブジェクト等）とが当たった（衝突ないし接触）か否かを判断する。この当たり検出は、図11のサブルーチンに従って実行される。

【0048】図11のステップS201において、CPU11は、 $ABS(OBJ2x - OBJ1x) \leq OBJ1r$ であるか否か、すなわち、X座標系で2つのオブジェクトが当たっているか否かを判断する。OBJ1は、当たり判定をする対象物のオブジェクトであり、この実施例では、プレイヤーオブジェクトのことである。OBJ2は、OBJ1に向かって近づいてくるオブジェクトであり、この実施例では、仲間オブジェクト、敵オブジェクト、静止オブジェクトおよび敵オブジェクトが発射した攻撃オブジェクトのことである。OBJ1xは、OBJ1のX座標値であり、OBJ2xは、OBJ2のX座標値である。OBJ1xとOBJ2xとは、同じ座標系のX座標値であれば、ゲーム空間座標であってもプレイヤー座標であってもよい。ABS()は、()内の数値の絶対値を表す。OBJ1rは、OBJ1を立方体として考えた場合の立方体の一辺の半分の長さを示す値である。換言すると、OBJ1rは、OBJ1の当たり判定範囲を示す値である。もし、 $ABS(OBJ2x - OBJ1x) \leq OBJ1r$ であるならば、ステップS202に進む。

【0049】ステップS202において、CPU11は、 $ABS(OBJ2y - OBJ1y) \leq OBJ1r$ であるか否か、すなわち、Y座標系で2つのオブジェクトが当たっているか否かを判断する。OBJ1yは、OBJ1のY座標値であり、OBJ2yは、OBJ2のY座標値である。OBJ1yとOBJ2yとは、同じ座標系のY座標値であれば、ゲーム空間座標であってもプレイヤー座標であってもよい。もし、 $ABS(OBJ2y - OBJ1y) \leq OBJ1r$ であるならば、ステップS203に進む。

J1rであるならば、ステップS203に進む。

【0050】ステップS203において、CPU11は、 $ABS(OBJ2z - OBJ1z) \leq OBJ1r$ であるか否か、すなわち、Z座標系で2つのオブジェクトが当たっているか否かを判断する。OBJ1zは、OBJ1のZ座標値であり、OBJ2zは、OBJ2のZ座標値である。OBJ1zとOBJ2zとは、同じ座標系のZ座標値であれば、ゲーム空間座標であってもプレイヤー座標であってもよい。もし、 $ABS(OBJ2z - OBJ1z) \leq OBJ1r$ であるならば、ステップS204に進む。

【0051】ステップS204において、CPU11は、OBJ2とOBJ1とが当たったと判断し、RAM14のフラグ領域142の当たりフラグF2を「1」にセットする。一方、ステップS201において、 $ABS(OBJ2x - OBJ1x) \leq OBJ1r$ でないならば、元のルーチンに戻り、ステップS202において、 $ABS(OBJ2y - OBJ1y) \leq OBJ1r$ でないならば、元のルーチンに戻り、ステップS203において、 $ABS(OBJ2z - OBJ1z) \leq OBJ1r$ でないならば、元のルーチンに戻る。

【0052】図10のステップS101において、プレイヤーオブジェクトが他のオブジェクトに当たっていないことを検出したとき、つまり、ステップS101において“NO”が判断されたとき、CPU11は、次のステップS102において、プレイヤーがコントローラ40を操作し、プレイヤーオブジェクトのアクセルをスタートさせたか否かを判断する。たとえば、「ウェーブレース」においてプレイヤーオブジェクトが「ジェットスキー」なら、Aボタン47A（図1）を押すことによって、また、プレイヤーオブジェクトが「マリオ」なら、ジョイスティック45（図1）を前に倒すことによって、アクセルスタートが実行される。したがって、CPU11は、このステップS102において、RAM14のコントローラデータ領域141のデータを参照して、Aボタン47Aやジョイスティック45が操作されたかどうかを判断する。

【0053】ステップS102において、“NO”が判断されると、CPU11は、次のステップS103において、プレイヤーオブジェクト（この実施例では、「ウェーブレース」の「ジェットスキー」）が水面に接しているかどうかを判断する。ステップS103において、プレイヤーオブジェクト（「ジェットスキー」）が水面に接しているかどうかを判断するために、図11の当たり検出サブルーチンが利用される。

【0054】このように、図10に示す振動発生条件検出サブルーチンにおいては、3つの振動発生条件すなわちステップS101、S102およびS103のいずれかを検出する。いずれのステップS101、S102およびS103においても“NO”が判断されると、すなわち、振動発生条件のいずれもが検出されないと、図1

0のステップS104において、CPU11は、RAM14のフラグ・レジスタ領域142のレジスタ値n__addを「0」にリセットする。それとともに、CPU11は、次のステップS105において、フラグ・レジスタ領域142の前フレームフラグF3を「0」にクリアする。すなわち、ステップS104においては、カウンタCTの増分値n__addを「0」に設定するため、カウンタCTのカウント値nは1フレーム毎に「0」加算される。換言すれば、全てのステップS101、S102およびS103において“NO”が判断されると、カウンタCTは全くインクリメントされない。後で述べるように、たとえばカウンタCTのカウント値が「255」を超えると振動が発生されるのであるから、この場合には、振動源507による振動は発生されない。

【0055】なお、先のステップS101において“NO”が判断されるため、ステップS105では、前フレームフラグF3がリセットされる。ステップS101、S102またはS103のいずれかにおいて“YES”が判断されると、振動発生条件が成立したので、それぞれの振動発生条件に従って、振動発生処理が実行される。

【0056】すなわち、ステップS101において、プレイヤーオブジェクトが他のオブジェクトと当たったことを検出すると、ステップS106ないしS110を実行し、振動源507から強い振動を発生させる。また、ステップS102において、アクセルスタートが検出されると、ステップS111ないしS113によって、弱い振動が発生される。プレイヤーオブジェクト（たとえば「ジェットスキー」）が水面に接していることをステップS103で検出すると、ステップS114ないしS117を実行し、「ジェットスキー」が波の上をバウンドする状態を表す弱い振動を発生させる。

【0057】先のステップS101において“YES”と判断されたとき、すなわち、プレイヤーオブジェクトが他のオブジェクトと衝突ないし接触したと判断したとき、CPU11は、前フレームフラグF3が「0」かどうか判断する。すなわち、このステップS106においては、前フレームにおいてもプレイヤーオブジェクトが他のオブジェクトと衝突していたかどうかを判断する。ステップS106において“YES”と判断されたとき、すなわち、前フレームではプレイヤーオブジェクトの衝突ないし接触はなく、現フレームにおいてプレイヤーオブジェクトと他のオブジェクトとの衝突ないし接触が検出されたとき、CPU11は、次のステップS107において、RAM14のフラグ・レジスタ領域142の増分値n__addを、強い振動を発生させるように、たとえば「255」に設定する。

【0058】つまり、図13に示すように、プレイヤーオブジェクトが敵オブジェクトに衝突ないし接触した場合、あるいは、図14に示すように、プレイヤーオブジェ

クトが敵オブジェクトないし静止オブジェクトに衝突ないし接触した場合には、強い振動を発生するために、ステップS107において、大きい増分値n__addを設定する。

【0059】次いで、ステップS108において、CPU11は、増分値n__addの変化値n__add_mを「20」に設定する。他のオブジェクトと衝突ないし接触したときには、強い振動を発生できるように、ステップS107において増分値n__addを「255」にするとともに、ステップS108において、変化値n__add_mを「20」に設定する。変化値n__add_mの値が大きいと増分値n__addが短い時間で「0」になり、変化値n__add_mの値が小さいと、増分値n__addが「0」になるまでに長い時間継続する。したがって、変化値n__add_mが大きいとき、振動源507による振動が長時間継続することになる。ステップS108に続くステップS109では、CPU11は、前フレームフラグF3を「1」に設定する。すなわち、現フレームにおいてプレイヤーオブジェクトの当たりが検出されたのであるから、その時点で前フレームフラグF3を「1」に設定するのである。

【0060】なお、先のステップS106において“NO”が検出されると、すなわち前フレームフラグF3が「1」であるとき、つまり、前フレームにおいてもプレイヤーオブジェクトの衝突ないし接触が検出されているときには、ステップS110において、増分値n__addを「0」にリセットする。すなわち、前フレームにおいてもプレイヤーオブジェクトが他のオブジェクトと接触ないし衝突したときには、継続的な振動の発生を防止するため、ステップS110において、増分値n__addを「0」とする。

【0061】ステップS102において、プレイヤーオブジェクトのアクセルスタートが検出されると、次のステップS111において、CPU11は、増分値n__addを、弱い振動を発生させるように、たとえば「150」に設定する。すなわち、プレイヤーオブジェクトのアクセルがスタートされたとき、先の当たり検出の場合と異なり比較的小さい振動を発生するために、カウンタCTの増分値n__addをステップS107より小さい「150」に設定する。そして、ステップS112において、CPU11が、変化値n__add_mを「10」に設定する。ステップS113において、前フレームフラグF3を「0」とする。すなわち、このステップS113は、先のステップS101において“NO”と判断されたとき実行されるステップであるから、先のステップS109とは異なり、前フレームフラグF3を「0」とする。

【0062】さらに、ステップS103において、プレイヤーオブジェクト、実施例においては「ジェットスキー」が水面に接していることが検出されると、次のステ

ップS114において、乗算値 f_x 、 f_y および f_z を計算する。そして、ステップS115において、CPU11は、カウンタCTの増分値 n_add として、「 $f_x \times f_x + f_y \times f_y + f_z \times f_z$ 」の平方根（小数点以下切り捨て）を計算する。つまり、このステップS114およびS115では、プレイオブジェクトのX軸方向、Y軸方向およびZ軸方向の加速度 a_x 、 a_y および a_z を求め、その加速度に比例した値 f_x 、 f_y および f_z を求め、この比例値ないし乗算値に応じて、

「0」～「255」の範囲の増分値 n_add を計算する。そして、ステップS116において、増分値 n_add の変化値 n_add_m として、たとえば「10」を設定する。すなわち、プレイオブジェクト「ジェットスキー」が水面に接している場合には比較的長時間衝撃を与えるために、変化値 n_add_m としては相対的に小さい「10」を設定する。そして、この場合にも先のステップS101において“NO”が判断されたのであるから、続くステップS117において、前フレームフラグF3を「0」にする。

【0063】振動発生サブルーチンを示す図12の最初のステップS120において、CPU11は、RAM14のフラグ・レジスタ領域142の振動ゲームフラグF1が「1」であるかどうか、すなわち、現在プレイ内のゲームが振動付加ゲームであるかどうかを判断する。そして、このステップS120において“YES”が判断されると、CPU11は、RAM14のコントローラデータ領域141を参照して、振動カートリッジ50（図1、図4）がコントローラ40に装着されているかどうかを判断する。振動ゲームでありかつ振動カートリッジ50がコントローラ40に装着されているとき、次のステップS122において、CPU11は、振動発生条件が成立したにも拘わらず振動を強制的に停止すべきかどうかすなわち振動がリセットされたかどうかを判断する。このステップS122において“NO”が判断されると、CPU11は、次のステップS123において、カウンタCTのカウンタ値 n を「 $n + n_add$ 」とする。すなわち、カウンタCTのカウンタ値 n を増分値 n_add に従って増加する。

【0064】そして、次のステップS124において、その増分値 n_add を、変化値 n_add_m に従って修正する。すなわち、このステップS124においては、ディスプレイ30の1フレーム毎に増分値 n_add の値を変化値 n_add_m だけ減算するため、カウンタCTのカウンタ値 n の値の増分値がフレーム毎に小さくなり、やがて、カウンタ値 n が増加しなくなる。換言すると、振動源507による振動は、最初は大きく（強く）かつ徐々に小さく（弱く）なり、やがて振動しなくなるのである。

【0065】そして、ステップS125においては、カウンタCTのカウンタ値 n が「255」を超えたかどう

かを判断する。このステップS125において“YES”が判断されると、カウンタCTのカウンタ値 n を「 $n - 256$ 」とする。すなわち、ステップS125において、カウンタCTのカウンタ値 n が「255」を超えたと判断されると、次のステップS126においては、そのカウンタ値 n から「256」を減算する。そして、ステップS127において、CPU11は、振動フラグF4を「1」に設定する。先のステップS125においてカウンタCTのカウンタ値 n が「255」を超えたことが検出されたので、このステップS127においては、振動フラグF4を「1」に設定する。

【0066】そして、次のステップS128においては、振動フラグF4が「1」かどうかを判断する。先のステップS127において、振動フラグF4が「1」に設定されていると、振動を発生するために、次のステップS129において、CPU11は、アドレスA15を除く全てのアドレスA2-A14に「1」を出力するとともに、書込信号を出力し、かつ、チップイネーブル信号を出力する。したがって、このステップS129において、レコーダないしNANDゲート511（図6）から信号が出力され、ラッチ512がCPU11のデータビットD0をラッチする。ステップS129においては、振動源507において振動を発生すべきであるので、CPU11のデータビットD0は「1」として出力される。したがって、ラッチ512（図6）には「1」がラッチされ、応じて、トランジスタ514がオンし、振動源ないし振動モータ507に電池504から電力が供給され、振動源507すなわち振動カートリッジ50、つまりコントローラ40が振動する。

【0067】なお、先のステップS125において“NO”が判断されると、ステップS130において、振動フラグF4が「0」にリセットされる。すなわち、カウンタCTのカウンタ値 n が「255」を超えないときには、振動フラグF4はリセットされたままである。そして、ステップS127において、振動フラグF4が「1」でないとき、すなわち振動フラグF4が「0」であるときには、ステップS131において振動を停止するために、CPU11は、データビットD0に「0」を出力する。したがって、ラッチ512に「0」がラッチされ、トランジスタ514がオフする。したがって、振動源507には電流が流れず、振動源507から振動が発生されることはない。

【0068】たとえば、プレイオブジェクトが敵オブジェクトや静止オブジェクトに衝突ないし接触した場合（図13、図14）、ステップS107において、カウンタCTのカウンタ値 n の増分値 n_add として「255」が設定されかつステップS108において、変化値 n_add_m として「20」が設定されるため、表1および図15に示すように、カウンタCTのカウンタ値 n が最初のフレームを除く4フレーム連続して「22

23

5」を超えるため、振動源507が4フレーム連続して駆動され、その後2フレーム毎にカウント値nが「255」を超え、振動源507は2フレーム毎に駆動される比較強い振動を発生する。

【0069】ところが、プレイヤオブジェクトのアクセラスタートのときには、比較弱い振動を発生させるために、ステップS111において増分値n_addとして「150」が設定され、ステップS112において、変化値n_add_mとして「10」が設定される。したがって、この場合には、表2および図16に示すように、2フレーム毎に、カウント値nが「255」を超えるため、振動源507は2フレーム毎に駆動され、そして、次の2フレームは振動を休止し、次の1フレーム振動源507を駆動した後、さらに、次の2フレーム振動を休止する。

【0070】すなわち、強い振動を発生する場合には、振動源507は数フレーム連続して駆動され、その後2フレーム毎に駆動されるように振動が漸減する。弱い振動を発生すべきときには、2フレーム毎に振動源507が駆動され、その後3フレーム毎に駆動されるように振動が漸減する。ただし、増分値n_addや変化値n_add_mすなわちこのような振動発生パターンは、任意に設定できることはいうまでもない。

【0071】なお、第3の振動発生条件すなわち、プレイヤオブジェクト（「ジェットスキー」）が水面に接している場合には、増分値n_addはプレイヤオブジェクトの加速度の関数として設定されるため、その加速度に応じて強い振動または弱い振動が発生される。

【0072】

【表1】

強い振動の例				
フレーム	n	n_add	n_add_m	振動信号
1	255	255	20	0
2	234	235	20	1
3	193	215	20	1
4	132	195	20	1
5	51	175	20	1
6	208	155	20	0
7	85	135	20	1
8	200	115	20	0
9	39	95	20	1
10	114	75	20	0
11	169	55	20	0
12	204	35	20	0
13	219	15	20	0
14	219	0	20	0
15	219	0	20	0
16	219	0	20	0
17	219	0	20	0
18	219	0	20	0
19	219	0	20	0
20	219	0	20	0

【0073】

【表2】

24

弱い振動の例				
フレーム	n	n_add	n_add_m	振動信号
1	150	150	10	0
2	34	140	10	1
3	164	130	10	0
4	28	120	10	1
5	138	110	10	0
6	238	100	10	0
7	72	90	10	1
8	152	80	10	0
9	222	70	10	0
10	26	60	10	1
11	76	50	10	0
12	116	40	10	0
13	146	30	10	0
14	166	20	10	0
15	176	10	10	0
16	176	0	10	0
17	176	0	10	0
18	176	0	10	0
19	176	0	10	0
20	176	0	10	0

10

20

30

40

50

【0074】なお、上述の実施例においては、図15または図16に示す強い振動または弱い振動を発生させるために、図12のフロー図を実行して、カウンタCTのカウント値nを増分値n_addや変化値n_add_mに従って変化させ、そのカウント値nが「255」を超えたとき振動フラグF4を「1」に設定して、振動源507を駆動するようにしている。つまり、先の実施例では、振動源507によって振動を発生させるかどうかはリアルタイムで計算しながら決定した。

【0075】これに対して、たとえば図17に示すように、RAM14のフラグ・レジスタ領域142に外部ROM21のプログラム領域22aから読み出した強振動パターンデータおよび弱振動パターンデータをそれぞれ記憶しておき、これらの振動パターンデータを選択的に読み出してそれに基づいて振動源507を制御するようにしてもよい。なお、強振動パターンデータとしては、先の表1におけるフレーム1～16のフレーム順次の振動信号「0111101010000000」を用いることが考えられる。また、弱振動パターンデータとしては、先の表2におけるフレーム1～16のフレーム順次の振動信号「0101001001000000」を用いることが考えられる。なお、必要に応じて、中振動パターンデータも記憶してもよい。

【0076】たとえば図10のステップS101でプレイヤオブジェクトの衝突ないし接触を検出したときには強振動パターンデータを読み出すと、最初のフレームでは振動源507は駆動されず、続く4フレーム連続して振動源507が駆動され、その後2フレーム毎に振動源507が駆動され、以後振動源507の駆動が停止される。たとえば図10のステップS102でアクセラスタートを検出したときまたはステップS103で水面との接触を検出したときには弱振動パターンデータを読み出す。したがって、最初の4フレームでは振動源507は2フレーム毎に駆動され、その後3フレーム毎に振動源507が駆動され、以後振動源507の駆動が停止され

る。

【0077】さらに、上述の実施例では、プレイヤーオブジェクトが他のオブジェクトと当たったとき等において、コントローラ40に装着された振動カートリッジ50の振動源507によって振動を発生させた。この振動と時間的に関連するタイミングで、ゲーム画像に視覚上の振動を付与するようにしてもよい。この場合、ディスプレイ30上でのゲーム画像の振動はコントローラ40の振動カートリッジ50によって生じる機械的振動より早く認識されるので、図9のステップS10で振動カートリッジ50による機械的振動を発生させた後、1ないし2フレーム以上遅れて、図9のステップS4またはステップS7によってゲーム画像に振動を付与すればよい。

【0078】たとえば図10のステップS101でプレイヤーオブジェクトの他のオブジェクトとの接触ないし衝突が検出されると、強い振動をゲーム画像上に発生させるために、ディスプレイ30の表示画面全体を振動させる。この場合、図9のステップS4のカメラ処理において、先に述べたカメラの視点座標を小刻みに変化させるようにすればよい。

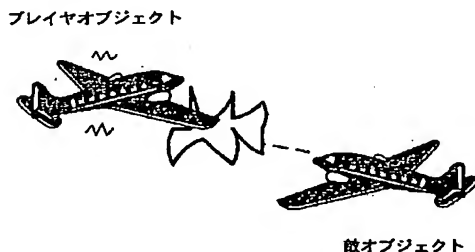
【0079】図10のステップS102またはS103において弱い振動を発生させるべき条件を検出すると、弱い振動をゲーム画像上に発生させるために、ディスプレイ30に表示されているプレイヤーオブジェクト画像にのみ振動を付与する。この場合、図9のステップS7の描画処理において、プレイヤーオブジェクトを形成する複数のポリゴン集合体の中心座標位置を小刻みに変化させるようにすればよい。

【0080】なお、ゲーム画像に振動を付与するかどうかは、図12のステップS128によって「F4=1」が検出されたかどうかを条件とするようにすればよい。つまり、振動フラグF4が「1」のときゲーム画像に振動を付与し、「0」のときにはゲーム画像の振動を発生させない。

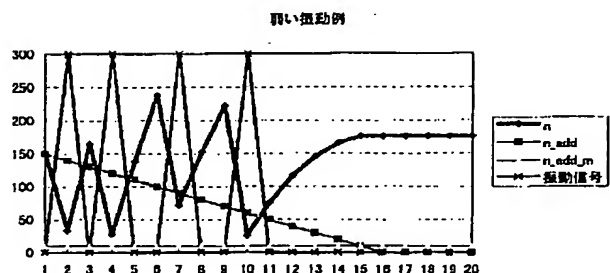
【図面の簡単な説明】

【図1】この発明の一実施例のビデオゲームシステムを示す概略図解図である。

【図13】



【図16】



【図2】図1のビデオゲーム機を詳細に示すブロック図である。

【図3】図2のコントローラ制御回路をより詳細に示すブロック図である。

【図4】図2のコントローラ制御回路および振動カートリッジを詳細に示すブロック図である。

【図5】振動カートリッジを詳細に示す図解図である。

【図6】振動カートリッジに含まれる振動発生回路を示す回路図である。

【図7】外部ROMのメモリマップを示す図解図である。

【図8】RAMのメモリマップを示す図解図である。

【図9】図1実施例の全体の動作を示すフロー図である。

【図10】振動発生条件検出サブルーチンを示すフロー図である。

【図11】当たり判定サブルーチンを示すフロー図である。

【図12】振動発生サブルーチンを示すフロー図である。

【図13】プレイヤーオブジェクトと敵オブジェクトとの衝突を示す図解図である。

【図14】プレイヤーオブジェクトと静止オブジェクトとの衝突を示す図解図である。

【図15】強く短い振動を発生させる振動パターンを示すグラフである。

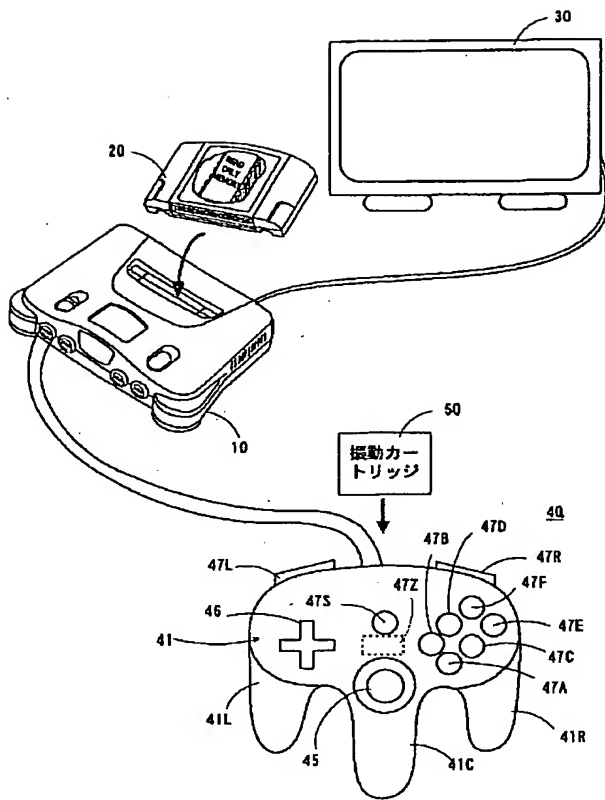
【図16】弱く長い振動を発生させる振動パターンを示すグラフである。

【図17】この発明の実施例におけるRAMのメモリマップを示す図解図である。

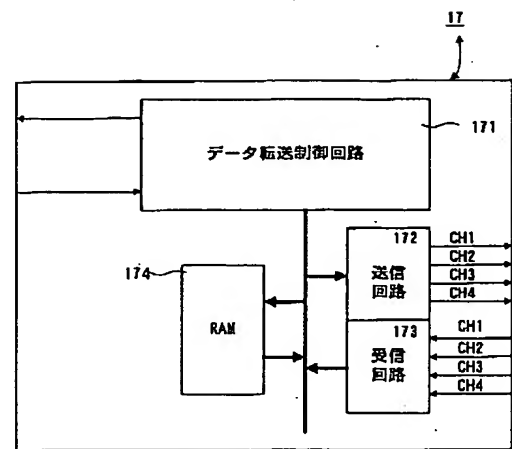
【符号の説明】

- 10 …ビデオゲーム機
- 11 …CPU
- 17 …コントローラ制御回路
- 20 …ROMカートリッジ
- 40 …コントローラ
- 50 …振動カートリッジ

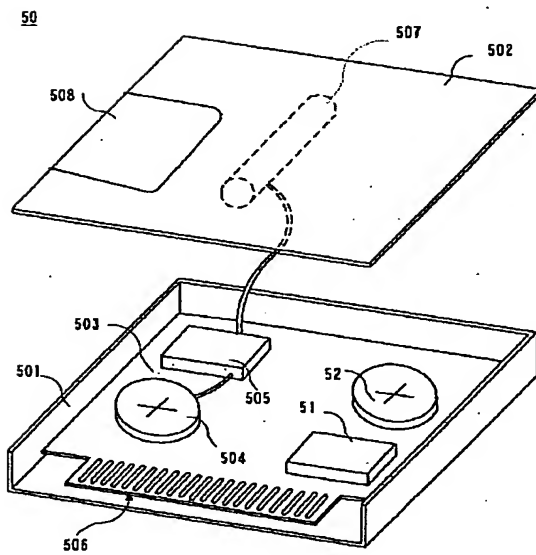
【図 1】



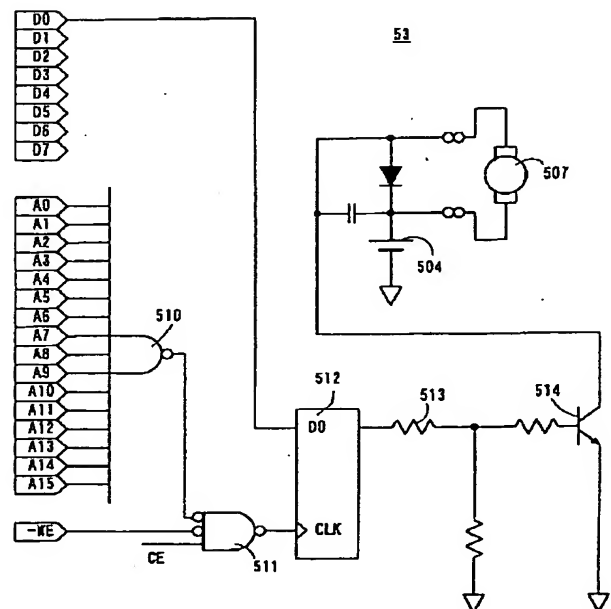
【図 3】



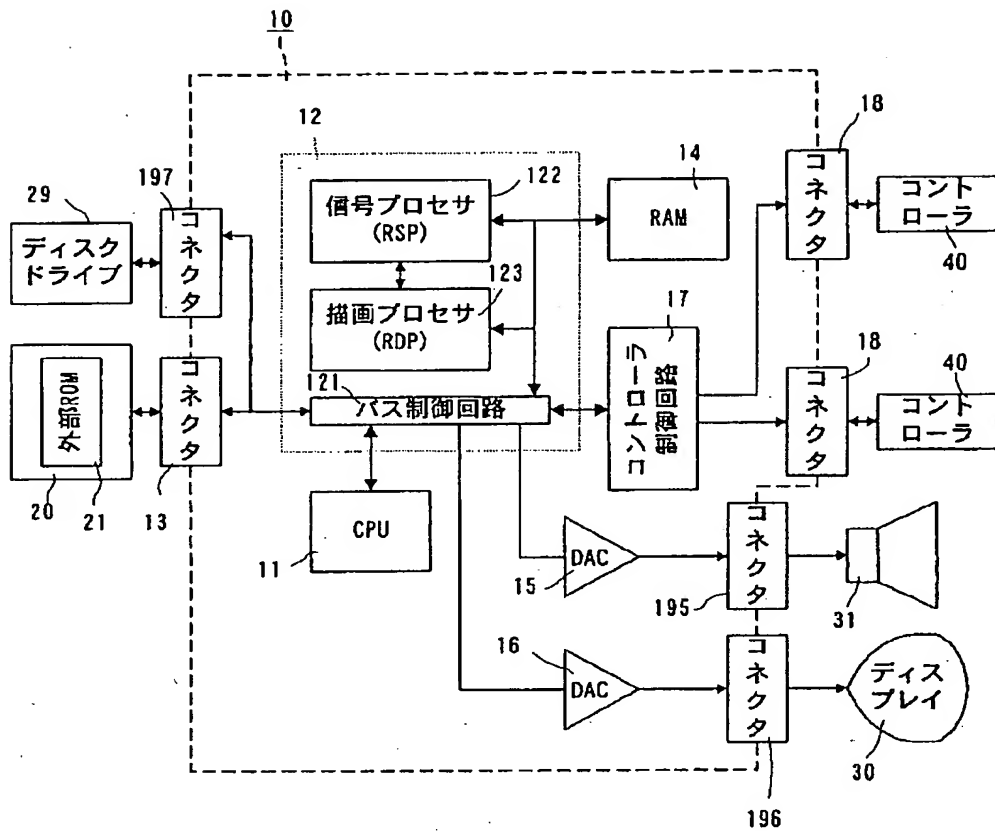
【図 5】



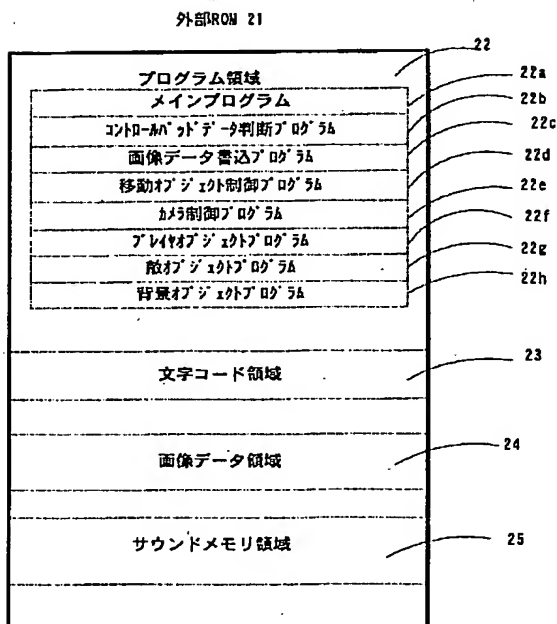
【図 6】



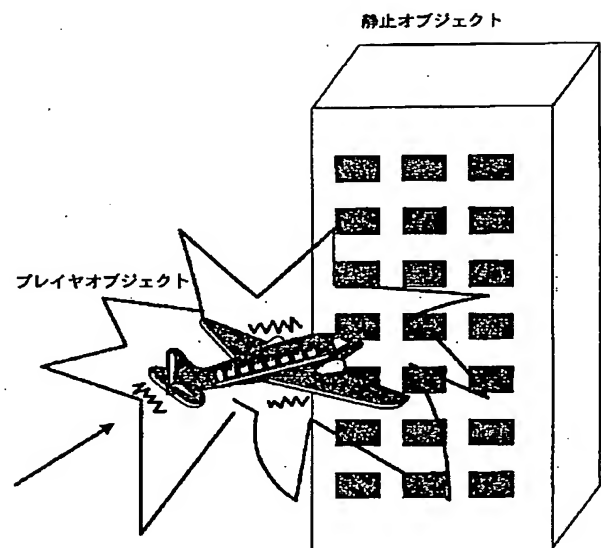
【図 2】



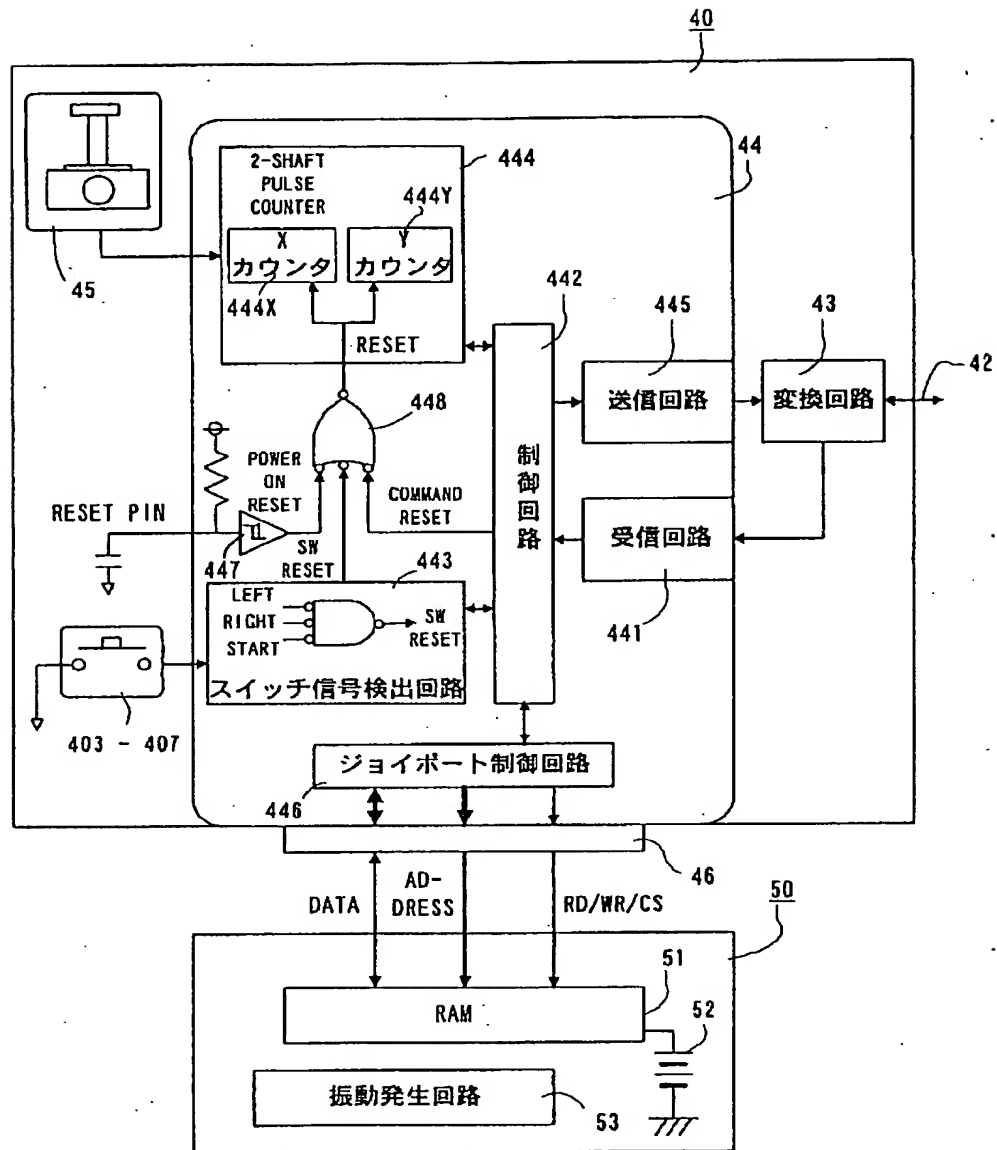
【図 7】



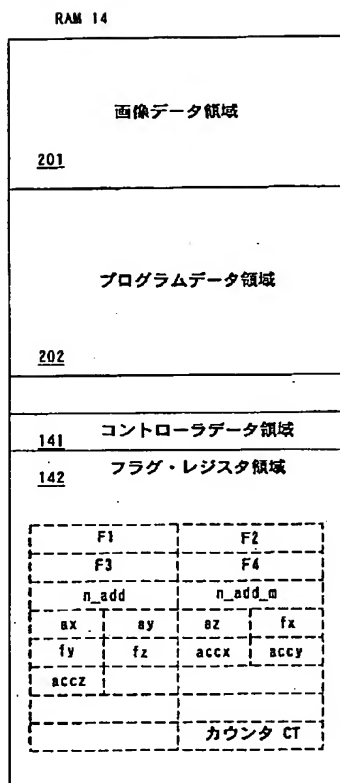
【図 14】



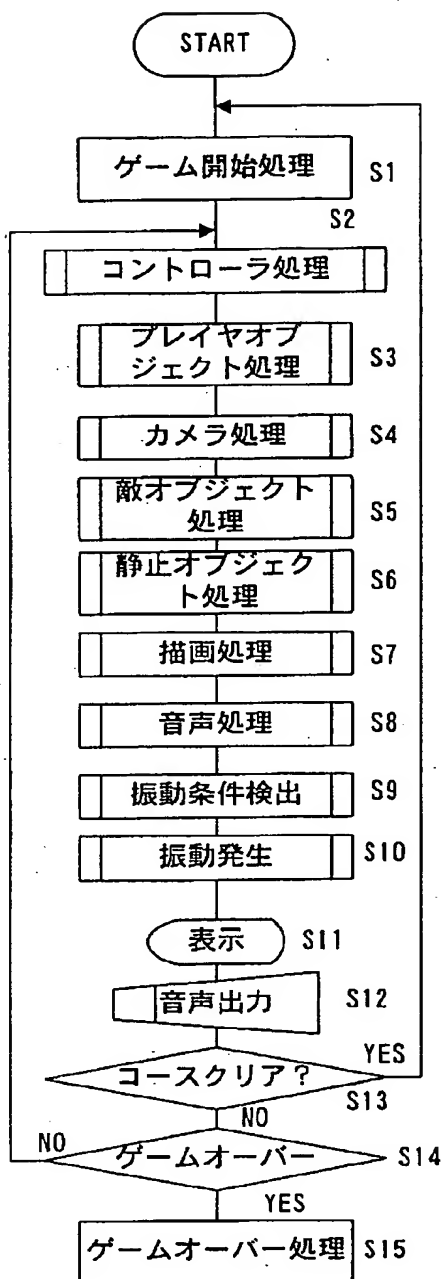
【図 4】



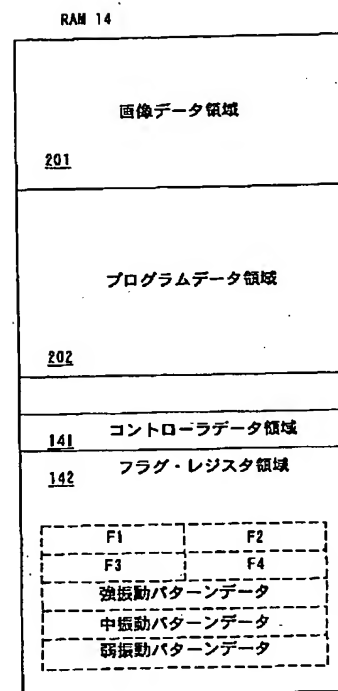
【図 8】



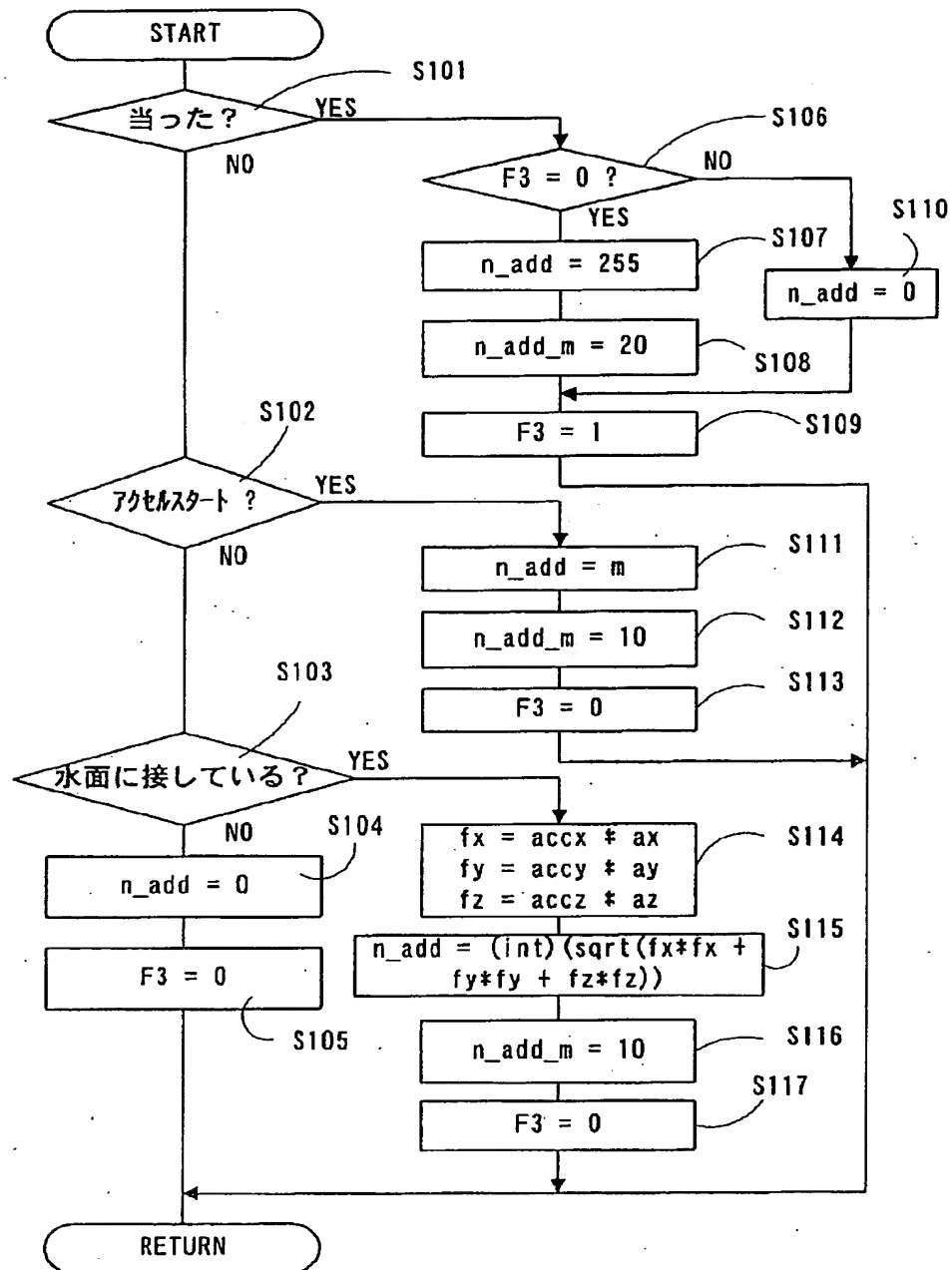
【図 9】



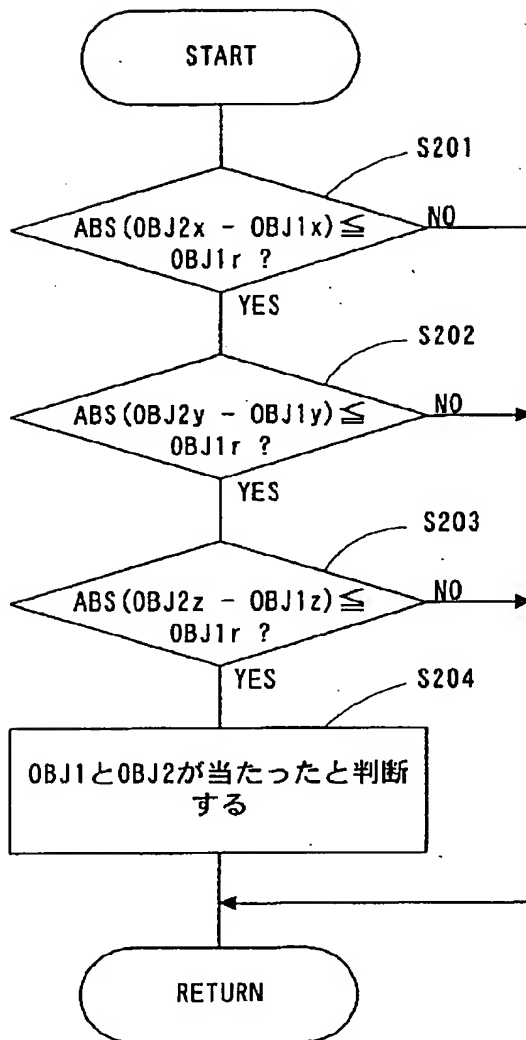
【図 17】



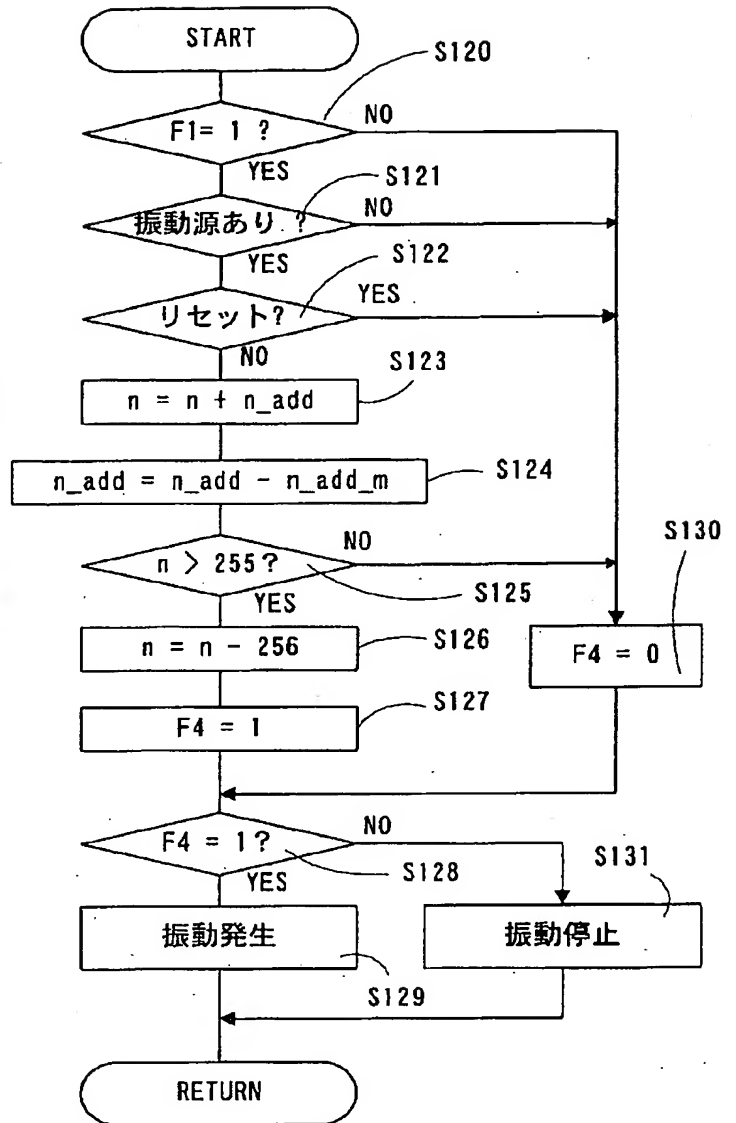
【図10】



【図 11】



【図 12】



【図 15】

強い振動例

